WHAT IS CLAIMED IS:

1. A processor comprising:

a register;

at least one writable store that, based on one or more values encoded therein, delimits a subset of memory addresses; and

a prefetch facility that initiates a prefetch based on correspondence of a value destined for the register with the delimited subset of memory addresses.

2. The processor of claim 1, further comprising:

logic responsive to an address calculation, the logic initiating the prefetch based on a match between the calculated address and the delimited subset of memory addresses.

3. The processor of claim 1, further comprising:

logic responsive to storage of a value into the register, the logic initiating the prefetch based on a match between the stored value and the delimited subset of memory addresses.

4. The processor of claim 1, further comprising:

address match logic that initiates the prefetch based on a match between the value destined for the register and the delimited subset of memory addresses, wherein the match is performed at a pipeline stage between address calculation and storage to the register.

5. The processor of claim 1,

wherein the value destined for the register is destined for an architectural register or any reorder buffer state corresponding thereto.

6. The processor of claim 1,

wherein the register is a register of an operative register set.

7. The processor of claim 6,

wherein the operative register set corresponds to a particular thread or process; and

wherein the delimited subset of memory addresses correspond to locations dynamically allocated by or for the particular thread or process.

8. The processor of claim 1,

wherein the at least one writable store includes a pair of registers whose contents delimit a contiguous range of memory addresses.

9. The processor of claim 1,

wherein, at runtime, the delimited subset covers a range of memory addresses that correspond to a heap.

10. The processor of claim 1,

wherein the prefetch is performed by the processor without presence of a corresponding prefetch instruction in an instruction sequence.

11. A processor that automatically prefetches data from memory based on detection, by the processor, of a likely pointer value destined for a register of the processor.

12. The processor of claim 11,

wherein the likely pointer value detection is based on correspondence of a calculated address with a predefined region of the memory.

13. The processor of claim 12,

wherein the predefined region of the memory is delimited by contents of at least one writable store.

14. The processor of claim 12,

wherein the predefined region of the memory is delimited by contents of at least a pair of registers that define a range of memory addresses.

15. The processor of claim 11,

wherein the likely pointer value detection is based on correspondence of a calculated address with a predefined region of the memory.

16. A method of automatically prefetching at least some data in a computer system, the method comprising:

executing an instruction sequence including a first instruction that targets a register; and

initiating, without a corresponding prefetch instruction in the instruction sequence, prefetch of data corresponding to a likely pointer value destined for the register as a result of the execution of the first instruction.

17. The method of claim 16, further comprising:

executing a memory access instruction that uses contents of the register as an address value, wherein prior performance of the prefetch allows the memory access instruction to be serviced from cache.

18. The method of claim 16, further comprising:

matching the likely pointer value against contents of at least one writable store that delimits a subset of addressable memory.

19. The method of claim 18,

wherein the at least one writable store includes a pair of registers that encode bounds of at least one contiguous portion of the delimited subset of addressable memory.

20. The method of claim 18,

wherein the delimited subset corresponds to a heap from which memory is dynamically allocated by or in the course of the instruction sequence.

21. The method of claim 20, further comprising:

initializing the at least one writable store to correspond to bounds of the heap.

22. The method of claim 18, further comprising:
initializing the at least one writable store to correspond to a range of memory addresses used for storage of lock states.

23. The method of claim 16, further comprising:
prefetching at least some other data based on a prefetch instruction in the instruction sequence.

24. The method of claim 16, further comprising:
prefetching at least some other data based on a prediction of memory access strides.

25. A method of operating a processor comprising:
detecting a likely pointer value destined for a register of the processor; and
prefetching from memory data corresponding to the likely pointer value.

26. The method of claim 25,
wherein the likely pointer value detection includes comparing against a predefined address pattern.

27. The method of claim 25,
wherein the likely pointer value detection includes comparing data values being stored into a register file against a predefined address pattern.

28. The method of claim 25,
wherein the likely pointer value detection includes scanning values stored in a register file or reorder buffer and comparing the scanned values against a predefined address pattern.

29. The method of claim 25,
wherein the processor supports speculative execution; and
wherein the likely pointer value detection includes comparing against a predefined address pattern, register states that become non-speculative.

30. The method of claim 25,

wherein the likely pointer value detection includes comparing against a predefined address pattern, address values calculated on execution of certain instructions of an instruction set.

31. The method of claim 26,

wherein the predefined address pattern is defined by contents of at least one writable store of the processor.

32. The method of claim 26,

wherein the predefined address pattern delimits one or more contiguous ranges of memory addresses.

33. The method of claim 26,

wherein the predefined address pattern delimits a subset of memory from which storage is dynamically allocated.

34. An apparatus comprising:

a processor; and

means for automatically prefetching certain data from memory based on detection of a likely pointer value destined for a register of the processor.

35. The apparatus of claim 34,

wherein the means for automatically prefetching includes at least one writable store of the processor, wherein contents of the at least one writable store delimit a range of memory addresses, and likely pointer value detection logic coupled to the at least one writable store and responsive to data values destined for register storage of the processor.

36. A method of making a processor that includes an automatic prefetch facility, the method comprising:

during fabrication of an integrated circuit, defining thereon at least one writable store of the processor suitable for delimiting a subset of addressable memory; and

during fabrication of the integrated circuit, defining thereon likely pointer value detection logic coupled to the at least one writable store and responsive to data values destined for register storage of the processor.